Parallel Implementation of the Wideband DOA Algorithm on the IBM Cell BE Processor

Todd E. Schmuland and Mohsin M. Jamali
Department of Electrical Engineering & Computer Science
The University of Toledo
Toledo, OH, 43606
todd.schmuland@utoledo.edu, mjamali@utnet.utoledo.edu

Matthew B. Longbrake and Peter E. Buxa
AFRL/RYDR
Wright-Patterson AFB
Dayton, OH, 45433
matthew.longbrake2@wpafb.af.mil, peter.buxa@wpafb.af.mil

***Abstract*—The Multiple Signal Classification (MUSIC) algorithm is a powerful technique for determining the Direction of Arrival (DOA) of signals impinging on an antenna array. The algorithm is serial based, mathematically intensive, and requires substantial computing power to realize in real-time. Recently, multi-core processors are becoming more prevalent and affordable. The challenge of adapting existing serial based algorithms to parallel based algorithms suitable for today's multi-core processors is daunting. One multi-core processor will be focused on, namely the IBM Cell Broadband Engine Processor (Cell BE). The process of adapting the serial based MUSIC algorithm to the Cell BE will be analyzed in terms of parallelism and performance for DOA determination.**

I. INTRODUCTION

Wideband digital processing of radar signals at a very high speed is a necessity for military and civilian applications. Furthermore, combining classification of radar signals with innovations in image processing to extract more information from data and linking to other intelligent databases will be the norm for future civilian and military applications. Often times these systems need to be small and mobile, but need to process more data. The latest computer system architectures developed by the computer industry offer more processing power in a smaller size, weight, and power footprint.

IBM has introduced a multi-core processor architecture called the Cell Broadband Engine (Cell BE) processor [1]. IBM recognized that traditional multi-core processors are being underutilized. Signaling and memory contention between the cores are issues that plague software writers when trying to parallelize algorithms to run on traditional multi-core processors. The Cell BE helps to alleviate these issues by providing an architecture that is easy to utilize with parallel algorithms.

This paper will focus on exploiting parallel processing and appropriately leveraging available multiple cores [2]. A wideband Coherent Signal-Subspace (CSS) Processing algorithm proposed by Wang & Kaveh [3] will be analyzed, parallelized, and mapped onto the Cell BE. The goal is to see whether the Cell BE can be used for real-time computation of Direction of Arrival (DOA) for wideband radar sources.

The Coherent Signal-Subspace (CSS) technique separates the wide frequency band into narrowband components [3]. The data set for this algorithm is divided into 64 segments and each segment contains 64 samples. We also assume a uniform linear array of 16 sensors. The Coherent Signal Subspace approach proposed by Wang & Kaveh [3] is used in this work and has the following computational steps:

- Compute 64 sets of 64-point FFT
- Compute 33 Covariance matrices (16 by 16)
- Computation of initial DOA estimate using Multiple Signal Classification MUSIC algorithm [4]
- Computation of Focus matrix
- Computation of number of sources
- Separation of Signal & Noise subspaces
- Compute DOA using MUSIC algorithm

Section II will cover unique features of the Cell BE. Section III will show the implementation of CSS algorithm on the Cell BE. Section IV will cover the simulation results with a conclusion in section V.

II. IBM CELL BROADBAND ENGINE PROCESSOR

The Cell BE is unique from traditional multi-core chips in that the nine cores that comprise the Cell BE are not all functionally the same, as shown in Fig. 1. One core, the Power Processing Element (PPE), is the master and runs typical PowerPC code. The other eight cores, the Synergistic Processing Elements (SPEs) can perform mathematical computation in parallel [1].

Each SPE unit has its own 256KB of memory that is loaded or stored from/to main memory via a Direct Memory Access (DMA) request to the Memory Transfer Engine (MTE). Once the local SPE memory transfer is complete, the SPE is free to use local memory for calculations without any data or timing conflicts with other SPE units.

This work was sponsored by the Dayton Area Graduate Studies Institute (DAGSI) with support from the Air Force Research Laboratory, Sensors Directorate, Wright Patterson Air Force Base, Ohio.

978-1-4244-5813-4/10/$26.00 ©2010 IEEE

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **MAY 2010**	2. REPORT TYPE	3. DATES COVERED **00-00-2010 to 00-00-2010**

4. TITLE AND SUBTITLE
Parallel Implementation of the Wideband DOA Algorithm on the IBM Cell BE Processor

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Toledo,Department of Electrical Engineering & Computer Science,Toledo,OH,43606

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM002322. Presented at the 2010 IEEE International Radar Conference (9th) Held in Arlington, Virginia on 10-14 May 2010. Sponsored in part by the Navy.

14. ABSTRACT
The Multiple Signal Classification (MUSIC) algorithm is a powerful technique for determining the Direction of Arrival (DOA) of signals impinging on an antenna array. The algorithm is serial based, mathematically intensive, and requires substantial computing power to realize in real-time. Recently multi-core processors are becoming more prevalent and affordable. The challenge of adapting existing serial based algorithms to parallel based algorithms suitable for today?s multi-core processors is daunting. One multi-core processor will be focused on, namely the IBM Cell Broadband Engine Processor (Cell BE). The process of adapting the serial based MUSIC algorithm to the Cell BE will be analyzed in terms of parallelism and performance for DOA determination.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**4**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

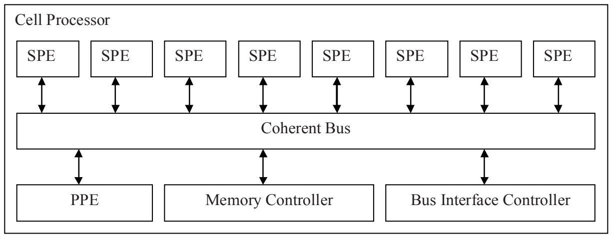

Figure 1. IBM Cell BE processor architecture.

The Cell BE provides novel methods for parallelism. Each SPE uses Single Instruction Multiple Data (SIMD) vector registers to operate on multiple operands during one instruction. Operands may be 8, 16, 32, 64, or 128 bits wide depending on the computation needed. With 128 vector registers, each 128 bits wide, four 32 bit operands are available to perform four calculations simultaneously. A simple communication method using mailboxes is used for communication between the SPE units and the PPE unit, thereby allowing SPE units to signal when they are done computing and have transferred their results back to main memory. The Cell BE includes DMA scheduling in hardware using priority levels and sequencing flags for automated queue management [1]. This allows the PPE to tell each SPE which of the eight DMA channels to use for its data transfers.

IBM and Sony have provided easy access to the IBM Cell Broadband Engine via the Playstation 3. This is an affordable vehicle for experimenting with parallel processing for sensor array processing applications [2]. A freely available simdmath library provided by Sony was used for efficient implementation of mathematical functions [5]. All of the mathematic functions use packed vector registers whereby four simultaneous calculations are performed in one function call. The Cell Processor has eight SPE cores, however not all eight are available for use in the Playstation 3. One SPE core is reserved as a proprietary hypervisor of the Playstation 3 hardware. To keep costs down, Sony accepts Cell BEs with one bad SPE core, which helps IBM with their chip yield rate. In fact, even those processors with eight good SPE cores have one SPE core permanently disabled before use in the Playstation 3. This results in six SPE cores available for use. The mailbox signaling technique will be used for communication between all six SPE cores and the PPE core during execution of this algorithm. Computations in all six cores will be in parallel and in a pipeline fashion.

III. PARALLELIZATION/ IMPLEMENTATION

The Cooley-Tukey based FFT is used to convert the sensor array data from the time domain to the frequency domain [6]. Due to symmetry, only half of the resulting frequency bins are required as covariance input. For example, a 64-point FFT will result in 32 unique frequency bins, ignoring DC. The FFT operation has been located inside the covariance module to maximize performance by eliminating unnecessary DMA transfers. An efficient bit reversal routine was used to perform the decimation-in-time swaps required. Vector registers are packed with data from the same row and column from four frequency bin matrices before being used by the covariance routine. This is slightly different than the narrowband covariance module in that four covariance matrices are produced simultaneously instead of only one.

To efficiently utilize the vector registers in the SPE units, the center frequency band is not used. For a 64-point FFT, this results in 32 frequency bins that map nicely to the four valued vector register format. So to compute the covariance for 32 frequency bins requires only eight separate matrix multiplications.

The MUSIC algorithm embedded in the wideband DOA algorithms [3-4, 7] utilizes the fact that the signal vectors are orthogonal to the noise subspace. To generate the noise subspace, the covariance of the FFT data vectors of samples obtained from the linear antenna array is computed. The covariance matrix is computed as $A^* \times A$, where A is the FFT data vector samples and A^* is the complex conjugate of A. The matrix A is in the complex domain so each element multiplication consists of $(a+bi) \times (c+di) = (ac-bd) + (ad+bc)i$.

The covariance matrix is computed by taking each row and multiplying it with each of the other rows' complex conjugates, itself included as shown in Fig. 2. Each element of the row is duplicated into all four locations of a vector register. Multiplication is then performed on all four values simultaneously for four consecutive rows of the matrix. The multiplication result is then accumulated for each element of the matrix. This achieves the $A^* \times A$ computation that results in the covariance matrix [3-4].

The local memory limitation of the SPE for calculation is handled by using mailboxes to issue memory pointers that point to the data to be acted upon. For example, the PPE uses a mailbox slot to send two memory pointers to a covariance module; one that points to the array sensor data and another that points where to store the results. Subsequently, that covariance module returns a message back to the PPE indicating when the covariance computation is complete.

Using the square covariance matrix, eigenvalue-eigenvector decomposition is performed and the resulting eigenvalue-eigenvector pairs are sorted in descending order according to the eigenvalues. The eigenvalue-eigenvector decomposition is performed by taking the covariance matrix and mapping it to the real domain. If C = A + Bi, where C is the covariance matrix, then R = [A –B; B A], where A and B

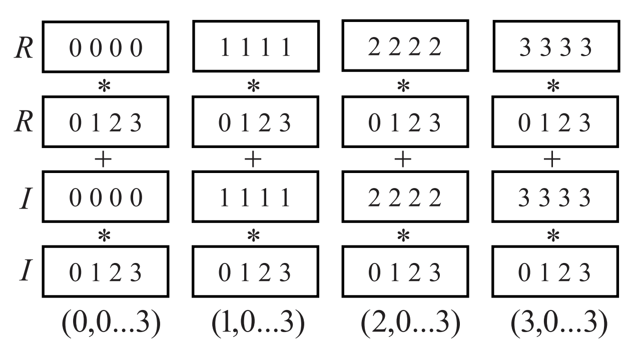

Figure 2. Covariance calculation of real part for the first four matrix elements of A. The imaginary part is similar except instead of RRII, the order is RIIR and subtractions are performed instead of additions.

are matrices consisting only of the real and imaginary parts respectively. The resulting matrix R is both real and symmetric.

The real-only symmetric covariance matrix is twice the width and height of the complex matrix it represents, however the complexity of the eigenvalue-eigenvector decomposition computation is significantly reduced. In addition, the eigenvalue-eigenvector decomposition method used in this paper only uses the lower triangle given to it, therefore the –B values can be skipped and a simple duplication of the A values is all that is required. This rearrangement is performed inside the eigenvalue-eigenvector decomposition module to prevent the duplicated A values from being transferred from system memory twice.

To maximize the performance, all available cores need to be utilized at near 100%. In addition, the overall bandwidth achieved hinges on how many sensor array samples can be included in the covariance computation in a given amount of time. Therefore, as Figs. 3 and 4 show, four SPE units were selected to perform covariance computations simultaneously, each on a separate block of sensor array data separated by time. The resulting four sets of 32 covariance matrices are then passed to the next cores for further processing.

The covariance matrix R is first reduced to tri-diagonal form using Householder transformations. The Householder transformation process is deterministic in nature and lends itself well to parallelization by vector register use. The next step is to reduce the tri-diagonal form to diagonal form using a combination of QL decomposition with implicit shifts and Givens rotations to maintain tri-diagonal form [6, 8]. The QL algorithm is not deterministic, so computation is terminated when convergence occurs within the limits of real number machine precision. Once the limit is reached, the resulting matrix is diagonal and represents the eigenvectors for matrix R.

Considerable effort was put forth into a parallel form of QL, however, the overhead complexity of tracking where in each matrix the Givens rotation should be performed, undermined the performance gain and introduced possible errors into the computation. In general, the diagonal decomposition converged in six iterations or less with the majority of iterations occurring in the upper half of the matrix. The lower half typically converged in one or two iterations. The convergence criteria was deemed true if the diagonal was within one epsilon unit of machine precision for single precision floating-point number representation. The result of the diagonal decomposition is a vector of eigenvalues with their respective normalized eigenvectors stored in matrix form. The eigenvalue-eigenvector decomposition is performed in the module labeled as EIG of Figs. 3 and 5.

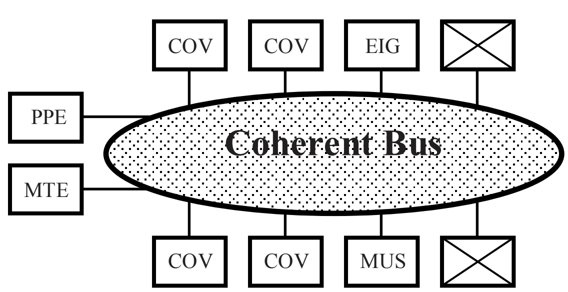

Figure 3. Mapping of SPE modules. Two SPE units are unavailable for use on the Playstation 3 platform.

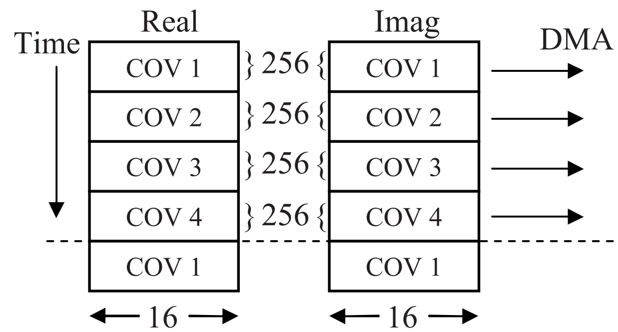

Figure 4. Partitioning of complex data samples for the sixteen sensor array with simultaneous DMA transfers to four covariance SPE units.

The last step of the eigenvalue-eigenvector decomposition is to sort the eigenvalues in descending order while keeping track of their respective eigenvectors. The eigenvector matrix is split into signal subspace and noise subspace using Akaike Information Criterion (AIC) [7]. The AIC criterion is used to determine the number of detected sources and separate the signal and noise spaces such that the number of signal eigenvectors matches the number of sources detected. The remaining noise space is then used as a guideline to detect the power peaks representing the detected signals and their respective angle of arrival. The number of signal sources is reported back to the PPE using the mailbox system. In addition, only the eigenvectors are returned by the eigenvalue-eigenvector decomposition module since the eigenvalues have served their purpose and are no longer needed.

The SPE unit labeled as MUS in Figs. 3 and 5 performs the computation of peaks using the power method, and lends itself well to parallelization. The power method is performed on the noise subspace eigenvectors for separate observations of the signals. The power is calculated as observed for each degree from 0 to 89 and returned back to shared system memory. The angles with the largest power peaks are the DOA of the incoming signals. It can be seen in Fig. 5 that all available cores of the Cell BE are all performing in parallel.

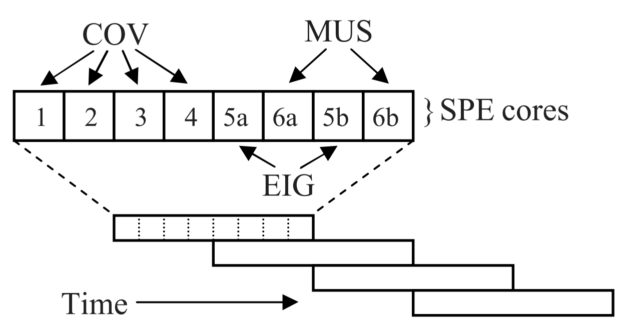

Figure 5. Pipeline stages of wideband DOA. The focusing is performed inside the 5b EIG after getting the initial DOA estimate from 6a MUS.

Together, 5a and 6a provide an initial DOA estimate for the wideband sources and are reused for final DOA determination.

This initial DOA estimate, along with the remaining frequency bin covariance matrices, is put through a focusing transformation inside the 5b EIG module followed by an eigenvalue-eigenvector decomposition. These eigenvalues and eigenvectors are again used in an AIC algorithm to estimate the number of sources. The number of detected sources is used to get the noise subspace. The power is calculated in 6b MUS as observed for each degree from 0 to 89 and returned back to shared system memory. The angles with the largest power peaks are the final DOA of the incoming signals.

IV. SIMULATION RESULTS

In order to demonstrate the parallel implementation of the DOA algorithm for wideband signals, a uniform linear array of sixteen equally spaced Omni-directional sensors was used. Two wideband sources at θ_1 and θ_2 were assumed. The signals are stationary zero mean band pass white Gaussian processes. Simulation data is similar to the method described by Wang & Kaveh [3].

The ease of signal detection using wideband DOA on a sixteen element sensor array is clear in Fig. 6. Four different FFT point sizes were tested and compared. The 64-point and 32-point FFTs show approximately the same results. Even the 16-point FFT gives adequate signal DOA detection, however the 8-point FFT starts to show extra peaks that could be misconstrued as valid signal sources.

Table 1 shows the bandwidth achieved using various approaches to the covariance module. The covariance computation time determines the number of possible sensor array samples processed over a given time and therefore the overall bandwidth achievable by the DOA algorithms.

A simple covariance calculation using single-threaded code running on the PPE can only provide a bandwidth of 11 kHz. Even though the Cell BE in the Playstation 3 runs at 3.2 GHz, the resulting bandwidth is very narrow.

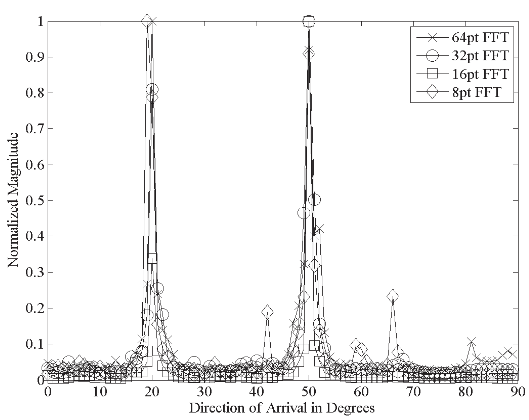

Figure 6. Wideband DOA of two incoming sources from 20 and 50 degrees using 4096 sensor array samples of a sixteen element array with four different FFT sizes.

TABLE I. IBM CELL BE MUSIC BANDWIDTH

Description	*Bandwidth*
Single-threaded narrowband MUSIC	11 kHz
Parallel Narrowband MUSIC	492 kHz
Parallel Narrowband unrolled loop	541 kHz
Parallel Wideband 8-point FFT	369 kHz
Parallel Wideband 16-point FFT	274 kHz
Parallel Wideband 32-point FFT	208 kHz
Parallel Wideband 64-point FFT	152 kHz

Parallel narrowband MUSIC when configured to utilize all six available cores has a respectable bandwidth of approximate 500 kHz. Parallel narrowband MUSIC with an unrolled inner loop can improve bandwidth by 50 kHz. This is done simply by unrolling the inner covariance multiplication loop, thereby leveraging many of the 128 vector registers available in the SPE unit. Parallel wideband DOA using various FFT transform lengths were computed and their bandwidth data is also given in Table 1. The 16-point FFT size demonstrates a good compromise between DOA resolution and bandwidth performance.

V. CONCLUSION

This paper focused on the feasibility of implementing a real-time parallel wideband DOA algorithm on the IBM Cell Broadband Engine processor. The Playstation 3 provided a quick parallel processing platform for this parallelization experiment. There are some hardware restrictions that have to be accounted for, such as limited local memory size for each SPE. However, by proper data arrangement to leverage the vector registers inherent in the SPE design, efficient modules for covariance, eigenvalue-eigenvector decomposition, and power method were successfully demonstrated.

REFERENCES

[1] IBM, Software development kit for multi-core acceleration version 3.1: Programmer's guide, Retrieved from http://publib.boulder.ibm.com/infocenter/systems/topic/eicct/prg

[2] J. Bartlett, Programming high-performance applications on the Cell BE processor, Retrieved from http://www-128.ibm.com/developerworks/power/library, 2007.

[3] H. Wang and M. Kaveh, "Coherent Signal-Subspace Processing for Multiple Wideband Sources," Proc. IEEE International Conference on Acoustic, Speech and Signal Processing, vol. 33, No. 4, pp. 823-831, August 1985.

[4] R. O. Schmidt, "Multiple emitter location and signal parameter estimation" IEEE Transactions on Antennas and Propagation, vol. AP-34, No. 3, pp. 276-280, March 1986.

[5] Sony Computer Entertainment, Inc. Retrieved from http://bullet.svn.sourceforge.net/viewvc/bullet/trunk/Extras/simdmathlibrary/

[6] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, Numerical recipes: The art of scientific computing, 3rd ed., Hong Kong: Golden Cup, 2007.

[7] H. Akaike, "A new look at the statistical model identification," IEEE Transactions on Automatic Control 19, 716-723.

[8] L. N. Trefethen, D. Bau III, Numerical linear algebra, Philadelphia: Society for Industrial and Applied Mathematics, 1997.